

# Method for discontinuous transmission, in sections, of data in a network of distributed stations

**Publication number:** DE10353564 (A1)

**Publication date:** 2005-06-16

**Inventor(s):** HUETTER INGO [DE]

**Applicant(s):** THOMSON BRANDT GMBH [DE]

**Classification:**

- international: **G06F13/00; H04L12/16; H04L29/08; H04N7/173; H04Q9/00; H04L29/06; G06F13/00; H04L12/16; H04L29/08; H04N7/173; H04Q9/00; H04L29/06; (IPC1-7): H04L29/08; H04N7/24**

- European: **H04L29/06M2H2; H04L29/08A7; H04L29/08N1; H04L29/08N5; H04L29/08N15**

**Application number:** DE20031053564 20031114

**Priority number(s):** DE20031053564 20031114

## Also published as:

EP1531601 (A1)  
EP1531601 (B1)  
US2005120132 (A1)  
MXPA04011196 (A)  
KR20050046567 (A)

more >>

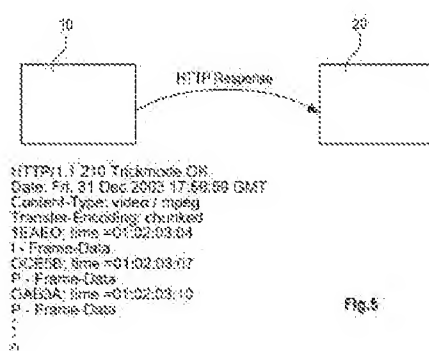
## Cited documents:

DE19543017 (A1)  
US6449653 (B2)  
US2003009537 (A1)  
US6604144 (B1)  
WO03028293 (A1)

Abstract not available for DE 10353564 (A1)

Abstract of corresponding document: **EP 1531601 (A1)**

When data streams are being transmitted in a network of distributed stations (10, 20) in which the network subscriber stations are controlled on the basis of the Internet Protocol, a resource (such as a file) can very often be transmitted using the HTTP-GET method. However, this does not support discontinuous transmission, in sections, of data, as is required, for example, in the case of trick modes (search processes) for a video film. The invention describes an extension to the known HTTP-GET method, such that this application is likewise possible. For this purpose, additional parameters relating to the required search are transmitted to the source appliance (10) in the HTTP-GET request. The source appliance (10) then sends the respective data sections for the search.





(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) **DE 103 53 564 A1** 2005.06.16

(12)

## Offenlegungsschrift

(21) Aktenzeichen: **103 53 564.0**  
(22) Anmeldetag: **14.11.2003**  
(43) Offenlegungstag: **16.06.2005**

(51) Int Cl.<sup>7</sup>: **H04L 29/08**  
**H04N 7/24**

(71) Anmelder:  
**Deutsche Thomson-Brandt GmbH, 78048**  
**Villingen-Schwenningen, DE**

(72) Erfinder:  
**Hütter, Ingo, 30982 Pattensen, DE**

(56) Für die Beurteilung der Patentfähigkeit in Betracht zu  
ziehende Druckschriften:

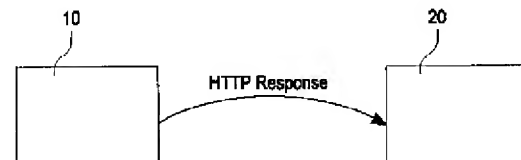
**DE 195 43 017 A1**  
**US 64 49 653 B2**  
**US2003/00 09 537 A1**  
**US 66 04 144 B1**  
**WO 03/0 28 293 A1**

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen**

Rechercheantrag gemäß § 43 Abs. 1 Satz 1 PatG ist gestellt.

(54) Bezeichnung: **Verfahren zur abschnittsweisen, diskontinuierlichen Übertragung von Daten in einem Netzwerk verteilter Stationen sowie Netzwerkteilnehmerstation als Anforderungsgerät bei der Durchführung eines solchen Verfahrens als auch Netzwerkteilnehmerstation als Quellgerät bei der Durchführung eines solchen Verfahrens**

(57) Zusammenfassung: Bei der Übertragung von Datenströmen in einem Netzwerk verteilter Stationen (10, 20), bei dem die Steuerung der Netzwerkteilnehmerstationen, basierend auf Internet-Protokoll, durchgeführt wird, ist es verbreitet, eine Ressource (wie zum Beispiel eine Datei) mit der HTTP-GET-Methode zu übertragen. Diese unterstützt aber keine abschnittsweise, diskontinuierliche Übertragung von Daten, wie sie bei Trickmodes (Suchlaufvorgängen) zum Beispiel bei einem Videofilm erforderlich ist. Die Erfindung beschreibt eine Erweiterung der bekannten HTTP-GET-Methode, so dass diese Anwendung ebenfalls möglich wird. Dazu werden bei der HTTP-GET-Anforderung zusätzliche Parameter, bezogen auf den angeforderten Suchlauf, zum Quellgerät (10) übertragen. Dieses sendet dann die jeweiligen Datenabschnitte für den Suchlauf in einem HTTP-Chunked Transfer-Encoding-Modus.



## Beschreibung

**[0001]** Die Erfindung betrifft das technische Gebiet der Datenübertragung in einem Netzwerk verteilter Stationen, insbesondere in einem sogenannten Heimnetzwerk. Dabei werden die Daten abschnittsweise und zusätzlich auch noch diskontinuierlich, d.h. nicht lückenlos übertragen.

### Hintergrund der Erfindung

**[0002]** Zur Vernetzung von Geräten im Heimbereich stehen mittlerweile unterschiedliche Heimnetzwerk-Standards zur Verfügung.

**[0003]** Ein Konsortium von Unternehmen, im besonderen die Unternehmen der Computer-Industrie, federführend ist Microsoft, haben eine Initiative für die Spezifizierung einer Netzwerksteuersoftware basierend auf dem existierenden Internet-Protokoll (IP) gestartet. Dieses Netzwerksystem ist unter dem Kürzel UPnP (Universal Plug and Play) bekannt geworden. Bei diesem System ist die Spezifizierung nicht primär für Unterhaltungselektronik-Geräte erfolgt, sondern es können in dem Netzwerk auch andere Geräte integriert sein, insbesondere Personal Computer, Haushaltsgeräte der weißen Ware, wie Kühlschränke, Mikrowellengeräte, Waschmaschinen, als auch Heizungssteuerungen, Lichtsteuerungen, Alarmanlagen und so weiter.

**[0004]** Für die Anwendung des UPnP-Verfahrens bei AV-Geräten wurde von einer Arbeitsgruppe des UPnP-Forums die UPnP-AV-Spezifikation erarbeitet, welche auf der allgemeinen UPnP-Spezifikation aufbaut und diese erweitert. Für den Transfer von AV-Daten (Audio/Video Daten) in einem solchen Heimnetzwerk zwischen einem sogenannten Server-Gerät (Quellgerät) und einem sogenannten Renderer-Gerät (Zielgerät) legt die UPnP-AV-Spezifikation fest, dass für den Transport der Daten bekannte Übertragungsprotokolle verwendet werden sollen. Als bekannte Protokolle werden in der Spezifikation die sogenannte HTTP-GET-Methode (HTTP steht für Hyper Text Transfer Protocol) und zum anderen das sogenannte RTP-Verfahren (RTP steht für Real Time Transport Protocol) genannt. Diese beiden Transportmechanismen stehen zur Verfügung, wenn die Netzwerkteilnehmerstationen über Ethernet-Busverbindungen mit einander verknüpft sind.

**[0005]** Das HTTP-GET-Verfahren basiert auf dem TCP-Verfahren (Transmission Control Protocol) welches ein grundlegendes verbindungsorientiertes Transportprotokoll ist, bei dem eine gesicherte Datenübertragung (mit Fehlerkorrektur) stattfindet. Das TCP-Verfahren wiederum baut auf dem Internet-Protokoll (IP) auf. Das HTTP-GET-Verfahren wurde speziell für die Übertragung von Dateien (z.B. HTML-Webseiten) von einem Webserver zu einem

Webbrowser entwickelt. Dadurch ist es nicht angepasst für die abschnittsweise Echtzeitdatenübertragung, wie sie z. B. bei der Übertragung von Audio- oder Videodatenströmen vorkommen. Andererseits ist das HTTP-GET-Verfahren weit verbreitet und für den Anwendungsprogrammierer sehr einfach gestaltet, so dass es sich großer Beliebtheit erfreut.

**[0006]** Der Transportmechanismus nach dem RTP-Verfahren basiert auf dem ebenfalls auf Internet-Protokoll aufbauenden UDP-Verfahren (User Datagram Protocol), welches verbindungslos arbeitet und keine Fehlerkorrektur verwendet, so dass bei Datenübertragungen nach diesem Verfahren Störungen auftreten können. Andererseits ist das RTP-Verfahren besser angepasst für Echtzeitdatenübertragungen, weil es Zwischenpuffer, Zusatzzeitstempel und Folgenummern verwendet. Für die Übertragung von Audio- oder Videoinformationen ist es daher insbesondere im durch zahlreiche Engpässe gekennzeichneten Internet in dieser Hinsicht besser geeignet.

**[0007]** Für die Übertragung von AV-Datenströmen wird in der UPnP-Spezifikation der Transportmechanismus nach der HTTP-GET-Methode empfohlen. Die HTTP-GET-Methode war primär dazu vorgesehen eine im Netzwerk vorhandene Ressource, das ist in vielen Fällen eine vorhandene Datei, anzufordern und dann vollständig in einem Stück zum Zielgerät zu übertragen. In der HTTP Version 1.1 wurde zusätzlich noch die Methode des sogenannten Chunked-Transfer-Encoding eingeführt, die immer dann zum Einsatz kommen soll, wenn eine Ressource zu übertragen ist, von der ihre Gesamtlänge zum Zeitpunkt des Übertragungsbeginns noch gar nicht fest steht. In diesem Fall soll eine abschnittsweise aber kontinuierliche (d.h. lückenlose) Übertragung der Ressource statt finden.

### Erfindung

**[0008]** Ausgehend von dem beschriebenen Stand der Technik ist es Aufgabe der Erfindung den Transportmechanismus nach der HTTP-GET-Methode so zu erweitern, dass auch die Realisierung von sogenannten Trickmodes bei der Übertragung von Datenströmen möglich wird. Solche Trickmodes sind z. B. bei AV-Datenströmen der schnelle Vorlauf (Suchlauf in Vorwärts-Richtung) und der schnelle Rücklauf (Suchlauf in Rückwärts-Richtung). Die Erfindung löst das Problem dadurch, dass für die HTTP-GET-Methode zusätzliche Parameter definiert werden, die z.B. die Abspielgeschwindigkeit und Abspielrichtung sowie die Anfangsposition für den Abspielvorgang betreffen. Für den Abspielvorgang bei solchen Trickmodes werden üblicherweise nur einzelne Datenblöcke benötigt, z. B. werden bei dem schnellen Suchlauf bei einem Videofilm während des Suchlaufs nur einzelne Videobilder wiedergegeben und andere Vi-

deobilder zwischen den wiedergegebenen Bildern unterdrückt. Es findet also praktisch ein wiederholtes Springen von einem Videobild  $N$  zu einem Videobild  $N + X$  (Vorlauf) bzw.  $N - X$  (Rücklauf) statt. Um dieses diskontinuierliche, abschnittsweise Übertragen der Daten zu realisieren, wird erfindungsgemäß im Rahmen der HTTP-Get Methode der Chunked-Transfer-Encoding-Mode abgewandelt und benutzt. Es wird nämlich immer ein wiederzugebender Datenabschnitt als ein Chunk übertragen. Dabei findet aber die Datenübertragung nicht kontinuierlich, also lückenlos statt, sondern zwischen den einzelnen übertragenen Datenabschnitten sind andere Bereiche, die ausgelassen werden, also handelt es sich um eine abschnittsweise, diskontinuierliche, Datenübertragung.

**[0009]** Mit dieser Erfindung wird es möglich den einfachen Transportmechanismus HTTP-GET auch für die Realisierung von Trickmodes bei der Echtzeitdatenübertragung, insbesondere für AV-Daten anzuwenden. Ebenfalls wird es mit dieser Erfindung möglich, sogenannte Navigationsbefehle zu implementieren, die z.B. einen gezielten Sprung an eine Position im Datenstrom erlauben, der eine gewisse Zeit, z.B. 15 Minuten später als die aktuelle Spielzeit liegt.

**[0010]** Durch die in den abhängigen Ansprüchen aufgeführten Maßnahmen sind vorteilhafte Weiterbildungen und Verbesserungen des erfindungsgemäßen Verfahrens möglich.

**[0011]** Als Rückmeldung, dass die Parameter in der erfindungsgemäßen HTTP-GET-Anforderung richtig interpretiert werden konnten, kann entweder ein vereinbarter Spezial-Statuscode zurückgeliefert werden oder aber es werden die Parameter in der Rückantwort noch mal wiederholt. Bei einem nicht erfindungsgemäß ausgelegten Gerät würde die Rückantwort anders ausfallen, so dass das Anforderungsgerät eine geeignete Information an die Bedierson ausgeben kann, wenn das angesprochene Gerät die Trickmode-Wiedergabe so nicht unterstützt.

**[0012]** Entsprechend vorteilhafte Maßnahmen für eine Netzwerkteilnehmerstation als Anforderungsgerät bei der Durchführung des erfindungsgemäßen Verfahrens sind in den Ansprüchen 10 und 11 aufgelistet.

**[0013]** Entsprechend vorteilhafte Maßnahmen für eine Netzwerkteilnehmerstation als Quellgerät bei der Durchführung des erfindungsgemäßen Verfahrens sind in den Ansprüchen 12 bis 16 aufgelistet.

#### Zeichnungen

**[0014]** Ausführungsbeispiele der Erfindung sind in den Zeichnungen dargestellt und werden in der nachfolgenden Beschreibung näher erläutert. Es zeigen:

**[0015]** Fig. 1 die groben Block-Schaltbilder eines Quell- und Zielgerätes in einem Netzwerk verteilter Stationen;

**[0016]** Fig. 2 das Format einer erfindungsgemäßen HTTP-GET-Anforderung, die vom Zielgerät zum Quellgerät gesendet wird;

**[0017]** Fig. 3 ein Beispiel einer Universalfernbedienung;

**[0018]** Fig. 4 eine schematische Darstellung, die als Beispiel einen Dreifach-Suchlauf in Vorwärts-Richtung bei MPEG2 kodierten Videodaten zeigt;

**[0019]** Fig. 5 das Format der erfindungsgemäßen HTTP-GET-Antwort;

**[0020]** Fig. 6 ein Ablaufdiagramm für ein Programm mit dem die HTTP-GET-Anforderung erstellt und zum Quellgerät gesendet wird;

**[0021]** Fig. 7 das Ablaufdiagramm eines Programms welches im Quellgerät abgearbeitet wird zur Erstellung und Aussendung der HTTP-GET-Rückantwort im Chunked-Transfer-Encoding Mode; und

**[0022]** Fig. 8 das Aussenden einer HTTP-GET-Anforderung vom Zielgerät zum Quellgerät mit Angabe einer absoluten URL, die durch Fragezeichen abgetrennt ebenfalls die erforderlichen Parameter für den angeforderten Abspielvorgang enthält.

#### Beschreibung der Erfindung

**[0023]** Fig. 1 zeigt ein Quellgerät **10** und ein Zielgerät **20** in einem Netzwerk verteilter Stationen. Wie dargestellt, sollen die einzelnen Netzwerkteilnehmerstationen durch Ethernet-Busverbindungen **15** miteinander verbunden sein. Weiterhin wird angenommen, dass die einzelnen Netzwerkteilnehmerstationen gemäß des UPnP-Standards ausgelegt sind. Die UPnP-Spezifikation kann über die Firma Microsoft bezogen werden.

**[0024]** Nähere Informationen sind auch auf der offiziellen Internetseite für das UPnP-System vorhanden. Diesbezüglich wird auf die Internetseite [www.UPnP.org](http://www.UPnP.org) hingewiesen.

**[0025]** Das Quellgerät entspricht in der Terminologie des UPnP-Systems einem Server-Gerät. Das Zielgerät **20** entspricht in der Terminologie des UPnP-Systems einem Renderer-Gerät, entsprechend Zielgerät. Im dargestellten Beispiel gemäß Fig. 1 sind für das Quellgerät **10** die zwei wesentlichen Komponenten Speichereinrichtung **12** und HTTP-Server **11** gesondert hervorgehoben. Typische Quellgeräte für Audio/Videodaten sind z. B. DVD-Spieler, DVD-Rekorder, Festplatten-Rekorder,

DVHS-Rekorder, CD-Spieler, MP3-Spieler, Videokamera oder auch ein Personal Computer. Die abspielbaren AV-Daten sind im Quellgerät **10** in der Speichereinrichtung **13** gespeichert. Die Komponente HTTP-Server **11** ist gemäß UPnP-Spezifikation zwingend für ein solches UPnP-Gerät vorgesehen. Die Komponente kann mit Softwaremitteln realisiert sein. Die entsprechenden Programm-Methoden sind aus dem Stand der Technik bekannt und es wird im Hinblick auf die Offenbarung dieser Komponenten insbesondere auf die UPnP-Spezifikation verwiesen.

**[0026]** Die wesentlichen Komponenten im Zielgerät **20** sind ein HTTP-Sende+Empfangsgerät **22**, HTTP-Server **21**, Pufferspeicher **23**, und eine Dekodierlogik **24**. Als Zielgeräte **20** kommen beispielsweise ein digitales Fernsehgerät, dieses kann auch in Form einer Set-Top-Box realisiert werden, ein DVD-Abspielgerät, ein MP3-Player und auch ein Personal Computer in Frage. Bei der Übertragung von MPEG2 kodierten Videoinformationen muss die Dekodiereinheit **24** als MPEG2-Dekoder ausgelegt sein. Der Pufferspeicher **23** soll vorzugsweise als RAM-Speichereinheit realisiert werden.

**[0027]** Im Folgenden wird angenommen, dass zwischen Quellgerät und Zielgerät eine eingerichtete Verbindung für eine Datenübertragung von AV-Daten besteht. Weiterhin wird angenommen, dass das Quellgerät **10** einen Videofilm mit Normalgeschwindigkeit abspielt und den zugehörigen AV-Datenstrom kontinuierlich über die eingerichtete Verbindung zu dem Zielgerät **20** überträgt. So dann fordert der Benutzer beim Zielgerät **20** einen schnellen Suchlauf in Vorwärts-Richtung an. Dies kann, wie in **Fig. 3** gezeigt, mit einer Fernbedienung **30** erfolgen. In **Fig. 3** ist als Beispiel eine Universal-Fernbedienung gezeigt, die mit Touch-Screen-Display **31** ausgestattet ist, auf dem die Bedienfunktionen für einen digitalen Videorekorder angezeigt sind. Solche Universal-Fernbedienungen sind aus dem Stand der Technik bekannt. Als Beispiel wird auf die ältere europäische Patentanmeldung EP-A-0 780 990 hingewiesen. Die auf dem Display **31** dargestellten Tasten sind eine Stopp-Taste **32**, eine Wiedergabetaste **33**, eine Pausentaste **34**, eine Aufnahmetaste **34**, eine Taste für schnellen Rücklauf **36**, eine Taste für schnellen Vorlauf **37**, eine Taste für einen Vorwärts-Titelsprung **38** sowie eine Taste für einen Rückwärts-Titelsprung **39**. Die Bedienperson hatte den schnellen Vorwärts-Suchlauf durch Drücken der Taste **37** beim Zielgerät **20** angefordert. Dieses Kommando wird dann vom Zielgerät **20** in eine entsprechende HTTP-GET-Anforderung umgewandelt und über das Netzkabel **15** zu dem Quellgerät gesendet. Die genaue Struktur der HTTP-GET-Anforderung ist in der **Fig. 2** angegeben. In dem dort gezeigten Beispiel steht nach dem Schlüsselwort GET der angeforderte Dateiname. Dies ist durch den Parameternamen ITEM signalisiert. Zum Beispiel lautet der

Dateiname MenInBlack.MPG was einem typischen Namen für einen Videofilm entspricht. Nach der HTTP-Versionsangabe in derselben Zeile folgt in der nächsten Zeile die Angabe in welchem Gerät diese Datei zu finden ist. Zur Kennzeichnung dieses Parameters ist in der zweiten Zeile der Ausdruck HOST dem Servernamen vorangestellt. Der im Beispiel angegebener Servername lautet www.my.server. In der darauf folgenden Zeile steht als nächster Parameter die Angabe der Abspielgeschwindigkeit. Der Parametername ist entsprechend AV\_SPEED. Dieser Parameter beinhaltet aber gleichzeitig auch die Abspielrichtung. Im dargestellten Beispiel besteht die Parameterangabe in dem Ausdruck Forward\_3 was bedeutet, dass der schnelle Suchlauf in Vorwärts-Richtung mit dreifacher Geschwindigkeit stattfinden soll. Für die Abspielrichtung könnte aber auch ein separater Parameter zusätzlich eingeführt werden.

**[0028]** In der vierten Zeile der HTTP-GET-Anforderung ist noch die Parameterangabe für die Anfangsposition des angeforderten Suchlaufs angegeben. Der Parametername hierfür ist AV\_STARTTIME. Nach dem Parameternamen steht dann die Positionsangabe 01.02.03.02. Dies gibt die aktuelle Spielzeitposition an, nämlich eine Stunde, 2 Minuten, 3 Sekunden, Bild 02. Für die Erstellung der HTTP-GET-Anforderung ist im Zielgerät **20** ein entsprechendes Programm vorgesehen, welches nach Anforderung des Suchlaufvorgangs (ausgelöst durch Drücken der Taste **37** abgearbeitet wird). Dieses wird nachfolgend noch näher erläutert.

**[0029]** In der **Fig. 4** ist jetzt der angeforderte Suchlaufvorgang mit dreifacher Abspielgeschwindigkeit in Vorwärts-Richtung näher dargestellt. Gezeigt ist die Darstellungsreihenfolge von digital kodierten Videobildern, wobei die digitalen Videobilder nach dem MPEG2-Standard kodiert sind. In diesem Standard wird zwischen intra-kodierten Bildern (I-Bilder) bidirektional-kodierten Bildern (B-Bilder) sowie unidirektional-kodierten Bildern (P-Bilder) unterschieden. Die höchste Datenkompressionsrate weisen die B-Bilder auf, gefolgt von den P-Bildern. Die intra-kodierten Bilder sind am wenigsten komprimiert, können dafür aber auch als Stütz-Bilder dienen, d.h. ihr Bildinhalt kann ohne Kenntnis eines vorhergehenden oder nachfolgenden Bildes rekonstruiert werden. Im MPEG2-Standard ist eine Abfolge von Bildern als sogenannte Group of Pictures GOP empfohlen. Diese lautet IBBPBBPBBPBB. **Fig. 4** zeigt drei aufeinander folgende solche Bildgruppen. Für die Realisierung des Dreifach-Suchlaufs in Vorwärts-Richtung wird jetzt wie folgt vorgegangen: Es werden nur die I-Bilder und P-Bilder der Bildgruppen wiedergegeben. Dadurch werden aus einer Bildgruppe mit 12 Bildern vier verschiedene Bilder dargestellt, so dass sich insgesamt ein Dreifach-Suchlauf in Vorwärts-Richtung ergibt. Bei dem Suchlauf wird also von I-Bild zu P-Bild, P-Bild zu P-Bild bzw. von P-Bild zu I-Bild im

Datenstrom gesprungen.

**[0030]** Für weitere Details zu der Technik, wie bei MPEG2 codierten Videobildern Trickmodes realisierbar sind, wird auf den Artikel „Entwicklung eines DVD-Players: Probleme und Lösungen“ von Ingo Hütter und Dirk Adolph in FKT Nr. 11/1999 Seiten 664ff verwiesen.

**[0031]** Fig. 5 zeigt das Format der HTTP-Rückantwort die vom Quellgerät **10** ausgegeben wird, nachdem der schnelle Suchlauf angefordert worden ist. In der ersten Zeile wird die Statusmeldung für eine erfolgreich eingetroffene HTTP-GET-Anforderung geliefert. Darin befindet sich auch die Versionsangabe bezüglich des HTTP-Verfahrens. In der zweiten Zeile der Rückantwort wird das Datum und die Uhrzeit angegeben.

**[0032]** In der dritten Zeile wird über das Schlüsselwort CONTENT-TYPE signalisiert, welcher Datentyp nachfolgend übertragen wird. Im genannten Beispiel handelt es sich um Videodaten, kodiert im MPEG2-Format. Mit der vierten Zeile wird dann signalisiert, dass die nachfolgende Übertragung im Chunked Transfer-Encoding-Mode stattfinden wird. Anschließend wird der erste Datenabschnitt gesendet. Vorangestellt ist die Längenangabe des betreffenden Datenabschnittes in Hexadezimalzahlen. Da wie zuvor im Zusammenhang mit Fig. 4 erläutert nur die Daten der I- und P-Bilder für den Suchlauf übertragen werden, folgen hier jeweils als ein Datenabschnitt die Videodaten eines I-Bildes. Für ein Standardvideobild beträgt die Datengröße eines solchen I-Bildes zirka 120 KByte. Dies schwankt natürlich von Bildinhalt zu Bildinhalt. Durch Semikolon abgetrennt von der Längenangabe folgt in derselben Zeile noch die jeweilige Zeitangabe für das übertragene Videobild. Der Suchlauf wurde mit der Anfangszeitangabe 01:02:03:02 angefordert, siehe Fig. 2. Das Quellgerät **10** hat ausgehend von dieser Zeitangabe das nächstfolgende I-Bild ausgesucht. Dieses wurde bei der Zeitposition 01:02:03:04 gefunden. Deshalb entspricht die Zeitangabe in der fünften Zeile der HTTP-Get-Rückantwort genau diesem Wert. In den nächsten Zeilen folgen dann die Videodaten für das angekündigte I-Bild. Nachdem dieses Videobild übertragen worden ist, folgt in der darauffolgenden Zeile die Längenangabe für das nächste zu übertragende Videobild mit der entsprechenden Zeitangabe an welcher Position im Videofilm dieses Bild lokalisiert ist. Die zugehörigen Videodaten folgen danach. Fig. 5 zeigt dann noch die entsprechenden Angaben für das dritte angesprungene P-Bild das danach übertragen wird. Der Vorgang setzt sich weiter fort, bis entweder der Suchlauf durch Benutzersteuerung beendet wird oder bis das Ende des Videofilms erreicht ist. Das Ende der Datenübertragung im Chunked Transfer-Encoding-Mode wird mit einer Zeile angezeigt, die lediglich als Wert den Wert 0 überträgt. Dieses ist

ebenfalls in Fig. 4 ersichtlich.

**[0033]** Wie beschrieben, handelt es sich um eine abschnittsweise aber diskontinuierliche Übertragung der Videodaten gemäß der Erfindung. Diskontinuierlich deshalb, weil nur die Videodaten einzelner Bilder, hier der I- und P-Bilder, übertragen werden, hingegen die Videodaten der dazwischen liegenden Bilder nicht übertragen werden.

**[0034]** Da es durchaus möglich ist, dass eine erfindungsgemäße HTTP-Anforderung an ein Gerät gesendet wird, das keine Trickmodi der beschriebenen Art unterstützt, sollte von einem Trickmodi unterstützenden Gerät in der Rückantwort deutlich gemacht werden, dass die Anforderung verstanden wurde und der Datenstrom konform zu den Parametern in der Anfrage ist. Dies könnte geschehen durch einen speziellen HTTP-Statuscode in der Rückantwort (z.B. „210“, „Trickmode OK“), wie in Fig. 5 gezeigt.

**[0035]** Alternativ kann in der Rückantwort auch eine Wiederholung der Trickmodeparameter erfolgen um Dieses anzuzeigen. Die Rückantwort wäre dann wie folgt aufgebaut:

```
HTTP/1.1 200 OK
Date: Fri, 31 Dec 2003 17:59:59 GMT
Content-Type: video/mpeg
Transfer-Encoding: chunked
AV_SPEED: forward_3
AV_STARTTIME: 01.02.03.04
1EAE0; time=01:02:03:04
```

**[0036]** In dieser Rückantwort wird einfach der bekannte Statuscode 200 OK verwendet. Aber dafür werden ja die Trickmodeparameter wiederholt, was kein Standard-Gerät machen würde, wenn es die Trickmodeparameter nicht interpretieren kann. Bei der Angabe des Parameters AV\_STARTTIME wurde die Zeitangabe in Übereinstimmung gebracht mit dem Wert, der zu dem ersten übertragenen Videobild gehört. Dies ist optional. Es kann auch einfach nur die Zeitangabe der Anforderung wiederholt werden.

**[0037]** Fig. 6 zeigt ein Beispiel eines Ablaufdiagramms für ein Programm das im Zielgerät **20** abgearbeitet wird, nachdem der schnelle Suchlaufvorgang vom Benutzer angefordert wurde. Das Programm startet mit dem Programmschritt **40**, nach dem die Bedienungsperson die Vorwärts-Suchlauffaste **37** betätigt hat. Da zuvor bereits ein Videofilm in Normalgeschwindigkeit abgespielt wurde, können im nachfolgenden Programmschritt **41** die erforderlichen Parameter für den Suchlaufvorgang noch von dem laufenden Videofilm extrahiert werden. Diese sind zum einen die Zeitangabe für die aktuelle Spielzeit, der Dateiname des wiedergegebenen Films, der Name des Servers auf dem der Videofilm gespeichert ist. Zusätzlich wird auch noch ein Parameter für den neuen Betriesmode erfasst. In einem einfachen Fall kann

dieser Parameter fest mit der Betätigung der Suchlauf Taste **37** verbunden sein. Dieser Taste ist dann fest der Parameter Vorwärts Suchlauf (dreifach) zugeordnet.

**[0038]** Im nächsten Programmschritt **42** wird die Normalwiedergabe des laufenden Videofilms gestoppt. Im anschließenden Programmschritt **43** wird der Pufferspeicher **23**, der noch Daten weiterer Videobilder enthalten kann, gelöscht. Durch diese Maßnahme wird erreicht, dass ein dedizierter Übergang von Normalwiedergabe zur Suchlaufwiedergabe stattfindet, und zwar ohne größere Verzögerungszeit, die gegebenenfalls als störend empfunden werden kann.

**[0039]** Im darauf folgenden Programmschritt **44** erstellt das Programm die nötige HTTP-GET-Anforderung für den schnellen Suchlaufvorgang. Das Format der zu erstellenden Anforderung ist in der **Fig. 2** gezeigt. Diese HTTP-GET-Anforderung wird in dem Programmschritt **45** an das Quellgerät **10** gesendet. Danach endet dieser Programmteil im Programmschritt **46**.

**[0040]** **Fig. 7** zeigt ein Beispiel eines Ablaufdiagramms für ein Programm das im Quellgerät nach Eingang der beschriebenen HTTP-GET-Anforderung abgearbeitet wird. Der Start des Programms ist mit Programmschritt **50** bezeichnet. Dieser wird durch Eingang des zuvor beschriebenen HTTP-GET-Requests ausgelöst. In dem Programmschritt **51** wird der eingegangene HTTP-GET-Request ausgewertet. Es werden die darin übertragenen Parameter insbesondere Anfangsposition für den neuen Abspielvorgang, File Name, Server Name, sowie Betriebsmode, d.h. Suchlauf mit welcher Geschwindigkeit in Vorwärts- oder Rückwärts-Richtung aus der HTTP-GET-Anforderung extrahiert. Das Beenden des laufenden Abspielvorgangs erfolgte automatisch nach Übertragung der HTTP-GET-Anforderung durch die bestehende TCP/IP-Verbindung bei dem Programmschritt **42**. Ausgehend von der übermittelten Ausgangsposition wird im nächsten Programmschritt **52** das darauf folgende I-Bild gesucht. Der neue Abspielvorgang startet dann bei dem so gefundenen I-Bild. Die dafür gelesenen Videodaten werden in eine HTTP-Antwortbotschaft eingebunden. Das Format der HTTP-Antwortbotschaft ist wie in der **Fig. 5** gezeigt. Es wird also der HTTP-Chunked Transfer-Encoding-Mode benutzt. Dieser Schritt ist im Programm der **Fig. 7** mit der Bezugszahl **53** bezeichnet. Nach dem die Videodaten des ausgewählten I-Bildes übertragen worden sind, springt das Laufwerk des Quellgerätes **10** die Stelle des nächsten darauf folgenden I-Bildes in der Videodatei an. Dafür sind gemäß MPEG2-Standard separate Zeigerinformationen im Videodatenstrom enthalten. Die Daten des nächsten I-Bildes werden dann als nächstes zu dem Zielgerät **20** übertragen. Dieser

Vorgang wiederholt sich fortlaufend so lange, bis entweder die Verbindung vom Zielgerät unterbrochen wurde oder bis zum Ende des Videofilms. Dann wird die in **Fig. 7** gezeigte Endlos-Schleife wird Durchbrochen.

**[0041]** Eine alternative Ausführungsform für eine HTTP-GET-Anforderung gemäß der Erfindung ist in der **Fig. 8** dargestellt. Erfindungsgemäß beinhaltet der darin gezeigte Pfadname nicht nur den Dateinamen, sondern auch alle weiteren erforderlichen Parameter für den neu zu startenden Suchlauf. Das Beispiel ist so ausgeführt, dass sämtliche erforderlichen Parameter in dem Pfad nach einem Fragezeichen aufgelistet sind. Gezeigt sind die Parameter Dateiname, Abspielgeschwindigkeit und Richtung sowie Anfangsposition. Mit dem Fragezeichensymbol können in einer URL (Uniform Resource Locator) eigene definierte Parameter durch &-Symbol getrennt an den Server übergeben werden. Dieses Verfahren ist in dem Internet Technologie-RFC 1738 näher beschrieben (RFC steht für Request For Comment).

**[0042]** Die Erfindung kann nicht nur für die Realisierung von Trickmodes (Suchläufen) bei der Übertragung von AV-Daten eingesetzt werden. Solche Suchläufe sind auch möglich bei anderen Multimediadaten. Als Beispiel wird eine Datei erwähnt, in der der Inhalt eines Buches gespeichert ist.

**[0043]** Weiterhin ist die Erfindung nicht nur auf den Anwendungsfall der Durchführung eines Trickmodes (Suchlauf) beschränkt. Damit lassen sich auch typische Navigationsvorgänge, wie zum Beispiel das Springen in Dokumenten an eine angegebene Position (z. B. an eine Position **15** Minuten später in dem Datenstrom realisieren). In diesem Fall würde in der HTTP-GET-Anforderung als Geschwindigkeitsangabe für die Wiedergabe Normalwiedergabe angegeben aber die neue Anfangsposition wäre entsprechend die Position **15** Minuten später im Datenstrom.

**[0044]** Weiterhin kann die Zahl und Art der erforderlichen Parameter je nach Ausführungsform variieren. Beispielsweise muss nicht immer die Anfangsposition als Parameter übergeben werden, wenn eine bestimmte Anfangsposition fest vorgegeben ist. Bei einem kurzen Beitrag kann vorgegeben sein, dass der Vorwärtssuchlauf jedes Mal von der absoluten Startposition erfolgen soll.

### Patentansprüche

1. Verfahren zur abschnittsweisen, diskontinuierlichen Übertragung von Daten in einem Netzwerk verteilter Stationen zwischen einer ersten (**10**) und zweiten Netzwerkteilnehmerstation (**20**), gekennzeichnet durch die folgenden Schritte:  
– Erstellen einer HTTP-GET-Anforderung mit Angabe der erforderlichen Parameter, wie Dateiname, Datei-

typ, Pfad, Abspielgeschwindigkeit, Abspielrichtung, Anfangsposition,  
 – Senden der HTTP-GET-Anforderung an das Quellgerät (10)  
 – abschnittsweises, diskontinuierliches Übertragen der Daten von dem Quellgerät (10) zu dem Zielgerät (20) im Modus des HTTP-Chunked Transfer-Encoding.

2. Verfahren nach Anspruch 1, wobei in der Rückantwort für die Übertragung der angeforderten Daten ein bestimmter Statuscode zurückgeliefert wird, der signalisiert, dass die erforderlichen Parameter in der HTTP-GET-Anforderung richtig interpretiert werden konnten.

3. Verfahren nach Anspruch 1 oder 2, wobei in der Rückantwort für die Übertragung der angeforderten Daten der/die erforderlichen Parameter von der HTTP-GET-Anforderung wiederholt werden, als Rückmeldung, dass die erforderlichen Parameter in der HTTP-GET-Anforderung richtig interpretiert werden konnten.

4. Verfahren nach einem der vorhergehenden Ansprüche 1 bis 3, wobei die abschnittsweise, diskontinuierliche Übertragung der Daten im Fall der Anforderung eines schnellen Suchlaufs eingesetzt wird.

5. Verfahren nach einem der Ansprüche 1 oder 2, wobei die Daten Multimediadaten betreffen, insbesondere Audio- oder Videodaten.

6. Verfahren nach einem der vorhergehenden Ansprüche, wobei zur abschnittsweisen, diskontinuierlichen Übertragung im HTTP-Chunked Transfer-Encoding-Modus, jeweils ein Abschnitt der Videodaten, insbesondere die Daten eines Videobildes, mit Angabe der Datenlänge als Block übertragen werden.

7. Verfahren nach Anspruch 6, wobei bei einem MPEG2-kodierten Videofilm jeweils die Daten der intrakodierten Videobilder oder der unidirektional kodierten Videobilder als ein Datenabschnitt übertragen werden.

8. Verfahren nach einem der vorhergehenden Ansprüche, wobei das Netzwerk verteilter Stationen ein auf Internet-Protokoll-Basis gesteuertes Netzwerk ist.

9. Verfahren nach Anspruch 8, wobei die Netzwerkteilnehmerstationen (10, 20) zur Steuerung nach der UPnP-Spezifikation ausgelegt sind, wobei UPnP für Universal Plug and Play steht.

10. Netzwerkteilnehmerstation als Anforderungsgerät bei der Durchführung des Verfahrens nach einem der vorhergehenden Ansprüche, gekennzeichnet

net durch Programmmittel zum Erstellen einer HTTP-GET-Anforderung mit Angabe der erforderlichen Parameter für ein abschnittsweises, diskontinuierliches Übertragen der Daten, wie Dateiname, Dateityp, Pfad, Abspielgeschwindigkeit, Abspielrichtung, Anfangsposition auf Anforderung hin.

11. Netzwerkteilnehmerstation nach Anspruch 10, mit Mitteln zum Empfang eines Befehls für die Durchführung eines schnellen Suchlaufvorgangs und Mitteln zum Aktivieren der Programmmittel zum Erstellen der HTTP-GET-Anforderung daraufhin.

12. Netzwerkteilnehmerstation als Quellgerät bei der Durchführung des Verfahrens nach einem der Ansprüche 1 bis 9, gekennzeichnet durch Programmmittel zum Auswerten einer empfangenen HTTP-GET-Anforderung von einem Zielgerät (20) für die abschnittsweise, diskontinuierliche Übertragung von Daten mit Angabe der erforderlichen Parameter wie Dateiname, Dateityp, Pfad, Abspielgeschwindigkeit, Abspielrichtung, Anfangsposition und Mitteln zum abschnittsweisen, diskontinuierlichen Senden der angeforderten Daten zu dem Zielgerät (20) im Modus des HTTP-Chunked Transfer-Encoding.

13. Netzwerkteilnehmerstation nach Anspruch 12, wobei im Fall der Übertragung von MPEG-Videodaten Mittel vorgesehen sind, die im Rahmen des HTTP-Chunked-Transfer-Encoding die Daten der intrakodierten Videobilder oder der unidirektional kodierten Videobilder des MPEG-Videodatenstromes als separate Datenabschnitte übertragen.

14. Netzwerkteilnehmerstation nach Anspruch 12 oder 13, wobei die Mittel zum abschnittsweisen, diskontinuierlichen Senden der Daten jeweils auch eine Positionsangabe hinzufügen, die die Stellung des Datenabschnitts im Datenstrom angibt.

15. Netzwerkteilnehmerstation nach einem der Ansprüche 12 bis 14, wobei Formatierungsmittel vorgesehen sind, die in der Rückantwort für die Übertragung der angeforderten Daten einen bestimmten Statuscode zurückliefern, der signalisiert, dass die erforderlichen Parameter in der HTTP-GET-Anforderung richtig interpretiert werden konnten.

16. Netzwerkteilnehmerstation nach einem der Ansprüche 12 bis 15, wobei Formatierungsmittel vorgesehen sind, die in der Rückantwort für die Übertragung der angeforderten Daten den/die erforderlichen Parameter von der HTTP-GET-Anforderung wiederholen, als Rückmeldung, dass die erforderlichen Parameter in der HTTP-GET-Anforderung richtig interpretiert werden konnten.

Es folgen 4 Blatt Zeichnungen



## Anhängende Zeichnungen

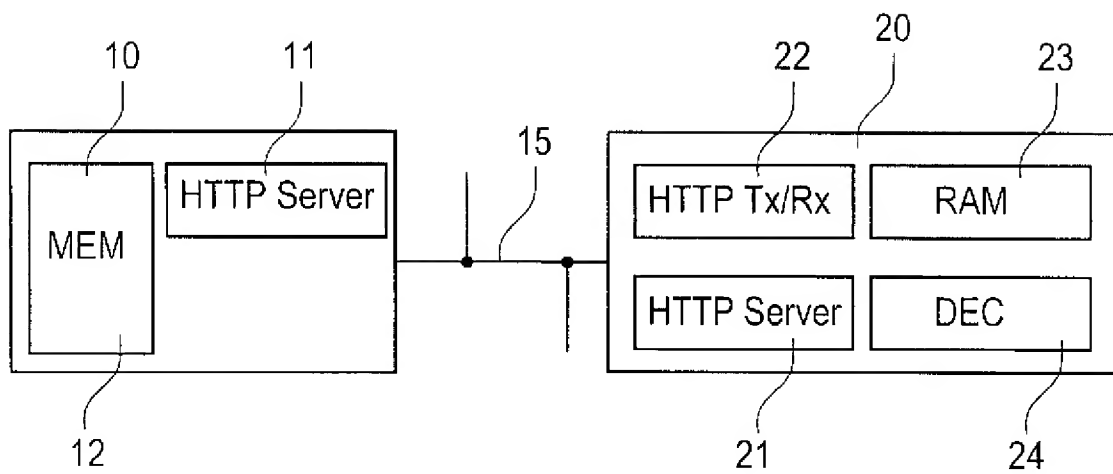
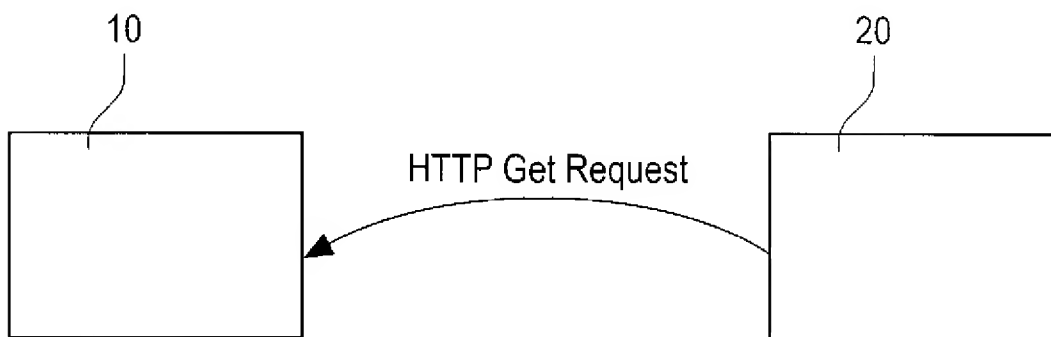


Fig.1



Get/item/MenInBlack.mpg HTTP/1.1  
 Host:www.my.server  
 AV\_SPEED: forward\_3  
 AV\_STARTTIME: 01.02.03.02

Fig.2

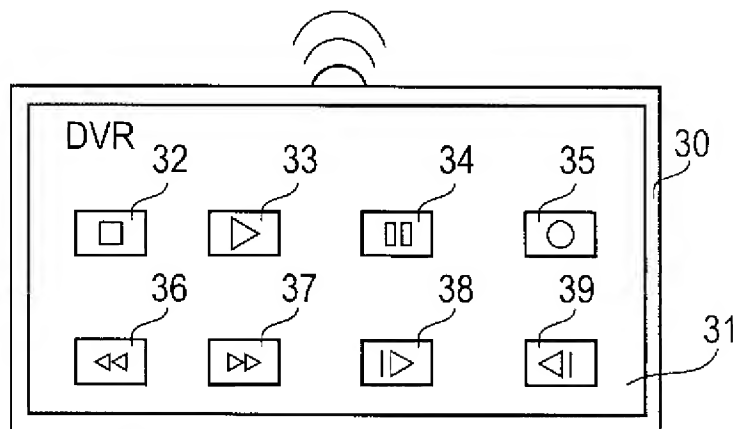
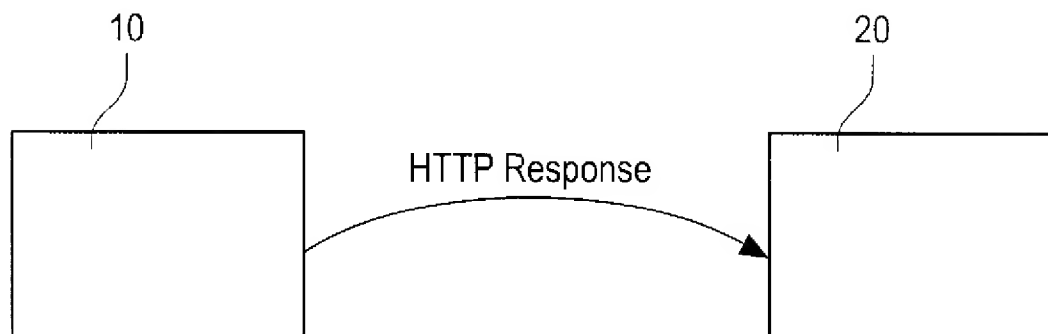


Fig.3

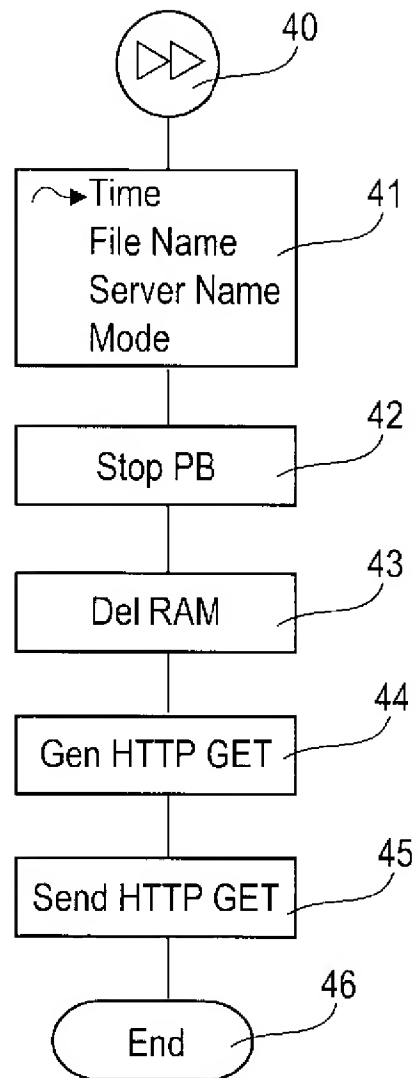
IBBPBBPBBPBB|IBBPBBPBBPBB|IBBPBBPBBPBB|B...  
| / / / / /  
IPPP|PPP|PPP...  
3 X FF

**Fig.4**

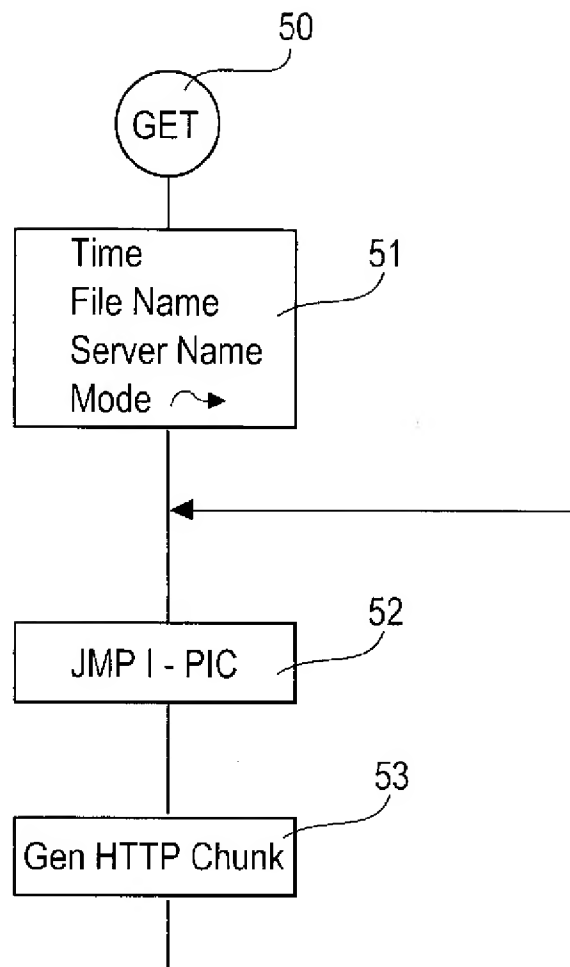


```
HTTP/1.1 210 Trickmode OK
Date: Fri, 31 Dec 2003 17:59:59 GMT
Content-Type: video / mpeg
Transfer-Encoding: chunked
1EAEO; time =01:02:03:04
I - Frame-Data
OCE5B; time =01:02:03:07
P - Frame-Data
OAB3A; time =01:02:03:10
P - Frame-Data
:
:
0
```

**Fig.5**



**Fig.6**

**Fig.7**

GET / item?name = MenInBlack&speed=forward\_3&time=01.02.03.02 HTTP/1.1  
HOST: www.my.server

**Fig.8**